11th World Congress on Computational Mechanics (WCCM XI)
5th European Conference on Computational Mechanics (ECCM V)
6th European Conference on Computational Fluid Dynamics (ECFD VI)
E. Oñate, J. Oliver and A. Huerta (Eds)

OPEN-WATER COMPUTATIONS OF A MARINE PROPELLER USING OPENFOAM

TUOMAS TURUNEN*, TIMO SIIKONEN*, JOHAN LUNDBERG†, AND RICKARD BENSOW††

* Aalto University
School of Engineering, PO Box 14400, Sähkömiehentie 4, 00076 Aalto, Finland,
e-mail: tuomas.a.turunen@aalto.fi

†Rolls-Royce Ab
Varnumsleden 5, 68193 Kristinehamn, Sweden,
e-mail: johan.lundberg@rolls-royce.com

††Chalmers University of Technology,
Dept. Shipping and Marine Technology, 41296 Göteborg, Sweden,
e-mail: rickard.bensow@chalmers.se

Key words: Moving Reference Frame (MRF), OpenFOAM, RANS, SST k-ω, Marine propeller, Open-water curve.

Abstract. The flow around a marine propeller is studied by means of a RANS-type finite-volume (FV) method. A performance curve in open-water conditions is reproduced computationally and compared to data measured in a cavitation tunnel at Rolls-Royce in Kristinehamn, Sweden. In the computations, both periodic grids with a single propeller blade and grids featuring a full propeller with five blades are used. Two types of full propeller grids are applied. The first one consists of tetrahedral control volumes with prismatic layers near the propeller surface. The second grid type is a hybrid mesh that is a combination of separately meshed hexahedral and tetrahedral grids with an Arbitrary Mesh Interface (AMI) between the two parts. Turbulence is modelled by the two-equation SST k-ω model as implemented in `OpenFOAM` .In steady-state computations, the Moving Reference Frame (MRF) approach is applied to account for the effects due to rotation. In case of transient simulations, moving meshes with the sliding plane approach are utilized and the effect of both numerical and iterative errors due to temporal discretization are studied.

This study provides a solid basis for more complicated simulations by providing an understanding of errors caused by grid interfaces and the choice of spatial and temporal discretizations of the governing equations. In particular, the experiences with the AMI methodology will allow the analysis of contra-rotating propellers (CRP) with `OpenFOAM`.

In addition, the robustness of the numerical method is considered in order to be able to produce a usable tool for industrial use.

1 Introduction

There is a need to make marine propellers more efficient. In order to improve current propellers and propulsion units, it is essential to possess the capability to analyze such applications. Current CFD methods include several techniques that can be applied for this task varying from simplified steady-state methods to complicated time accurate solvers. There are also several pieces of software available with these capabilities.

In this study, methods available in the open-source CFD software package `OpenFOAM` are tested. The software is licensed and distributed by the OpenFOAM Foundation and developed by OpenCFD Ltd [2]. `OpenFOAM` allows creating tailored computational tools and has no licencing costs. All computations aim at gaining more knowledge about the performance of the methods. Not only propeller forces are important but attention is also paid to the prediction of propeller wake in order to evaluate the feasibility of the methods for future studies with more than one propeller.

2 Numerical Method

The flow is predicted by enforcing the conservation of mass and momentum. These conservation equations are commonly known as the Navier-Stokes equations and they are used in their incompressible form. Turbulence is modelled with the two-equation SST k-ω model as implemented in `OpenFOAM 2.2.0` [1].

All the steady-state computations in this study are based on the Semi-Implicit Method for Pressure Linked Equations (SIMPLE) algorithm. The methods are as they are implemented in `OpenFOAM 2.2.0`. In the steady-state solver `simpleFoam` the flow is solved in a relative coordinate system that follows the blade movement according to the Moving Reference Frame (`MRF`) method. The original Navier-Stokes equations are modified so that convection is handled by the relative velocity and an additional source term is included in the equations.

The transient code, `pimpleDyMFOAM`, solves the time-accurate Navier-Stokes equations through an iteration procedure within each time step. This also enables the usage of under-relaxation in the transient computation, which improves stability and allows a larger time step. Propeller movement is realized by moving a part of the mesh around the propeller geometry. Since there is a discontinuity in the grid, an interpolation method is needed for the communication between both sides of the discontinuity. This method is commonly known as the sliding grid approach, which is called the Arbitrary Mesh Interface (AMI) in `OpenFOAM`. The same method is used in the steady-state computations. For more details on the AMI, the reader should refer to [5].

3 Computational Grids

Two grids are used in this study. The first one only comprises one blade and has periodic boundaries. The computational domain is a cylindrical volume with the propeller and hub in the center of it. The propeller and the hub are shown in Figure 1. The blue colour illustrates the parts that are included in the periodic grid. The computational

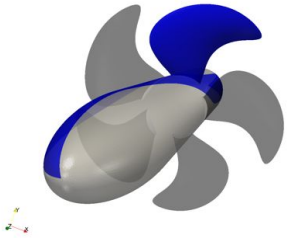
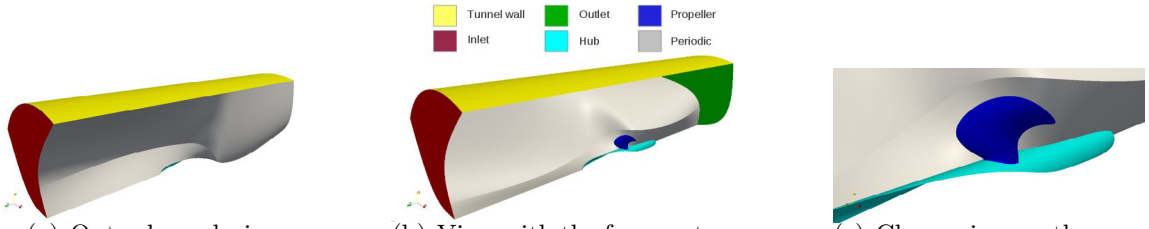


Figure 1: The complete propeller with the part included in the periodic grid colored in blue.

domain consists of a volume around one blade that follows the shape of the blade so that boundaries are kept as far from the blade as possible. The complete computational domain is shown in Figure 2. The domain stretches approximately four propeller diameters upstream and downstream of the propeller and the cylinder-like outer boundary is approximately two propeller diameters from propeller axis. Propeller diameter is $D \approx 25$ cm and the number of revolutions per second is $n \approx 17\ \frac{1}{\mathrm{s}}$. The blade surface mesh is refined at the leading and trailing edges where stronger gradients are expected. The surface mesh is presented in Figure 3. The cell size normal to the blade surface is such that the first cell y^+ value is approximately 1. The periodic grid consists of a total of 3 Mio. cells.



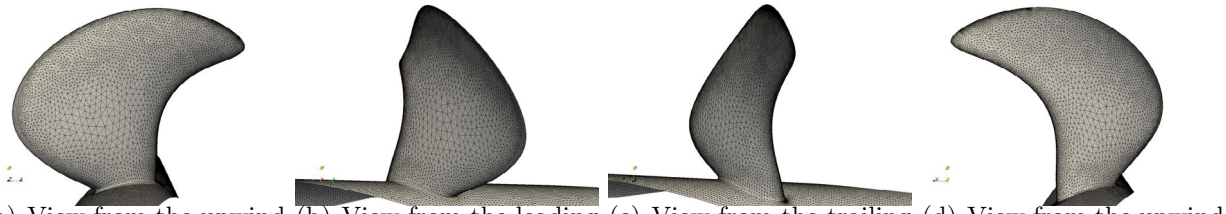
(a) Outer boundaries

(b) View with the foremost periodic boundaries hidden

(c) Closer view on the propeller and hub

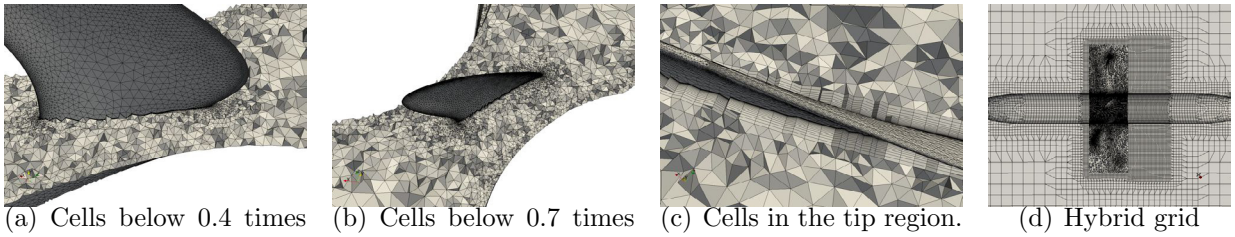
Figure 2: Case set-up in the steady-state computations with one blade only. Boundary names are given in part b).

The second model is a hybrid grid that consists of two separately meshed parts. The propeller region is meshed in the same way as the periodic grid but it is much smaller than the first grid. The propeller mesh consists of tetrahedral cells everywhere except at the vicinity of the blade where 20 layers of prismatic cells are used to predict the boundary



(a) View from the upwind direction (b) View from the leading edge (c) View from the trailing edge (d) View from the upwind direction

Figure 3: Surface grid on the propeller surface.



(a) Cells below 0.4 times the propeller radius (b) Cells below 0.7 times the propeller radius (c) Cells in the tip region. A closer view. (d) Hybrid grid

Figure 4: Volume mesh around the propeller.

layer more accurately. The surrounding part is meshed with the `snappyHexMesh` utility that is included in the `OpenFOAM` distribution and the two parts are then merged with mesh manipulation tools available in `OpenFOAM`. `snappyHexMesh` produces hexahedral cells that are quite efficient. The volume mesh is illustrated in Figure 4. The interface between the propeller and the surrounding parts is clearly visible in the last subfigure.

4 Results

4.1 Steady-State Computations

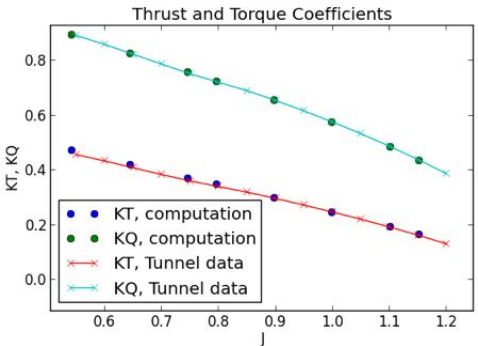
An open-water performance curve was computed with the periodic grid. The discretization of the convection term in the momentum equation was `limitedLinear 0.33`. The scheme refers to the method that is used to interpolate convected velocities onto cell faces. In the `limitedLinear` scheme the interpolation depends on the flow field itself. For more information, the reader should refer to the `OpenFOAM` documentation and source code.

The performance data are given the form of dimensionless propeller thrust and torque, K_T and K_Q, plotted against the dimensionless ship speed called advance coefficient, J. The dimensionless quantities are defined as

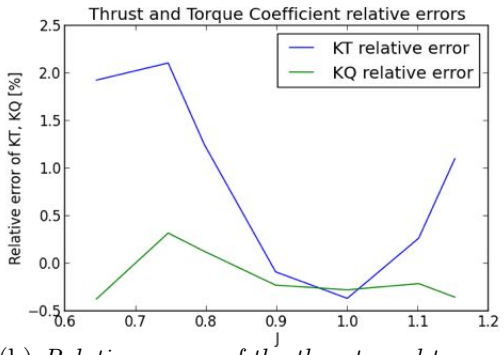
$$K_T = \frac{T}{\rho n^2 D^4} \ , \qquad K_Q = \frac{Q}{\rho n^2 D^5} \ , \qquad J = \frac{V}{nD} \tag{1}$$

where ρ is the water density, n the number of propeller rotations per second and D the propeller diameter. The dimensional quantities are propeller thrust T, propeller torque

Q and ship speed V [3].



(a) *Absolute values of K_T and K_Q as compared to experimental data over a range of advance coefficient values J.*



(b) *Relative errors of the thrust- and torque coefficients relative to measurements over a range of advance coefficient values J.*

Figure 5: Open water performance.

The results are shown in Figure 5. The lines correspond to data that has been measured at Rolls-Royce in Kristinehamn, Sweden, and the dots correspond to results from the computations. The force prediction is very close to the measured data which is somewhat surprising. The measument facility is not identical to the computational set-up. The biggest difference is the hub geometry. In the computations, the hub has a cigar-like form while in the experiments the hub continues further upstream. Due to the different upstream geometries, the incoming flow should differ between the computations and measurement which should also affect the force prediction. Furthermore, there are simplifications in the computational model such as the turbulence model and the steady-state approximation, which differ from real world physics. Thus, it is not quite clear exactly how large the errors are with respect to reality. More propellers should be tested in order to gain more exact information in this respect. The results, however, show that the overall set-up such as the boundary conditions and numerical settings are set correctly. Also the correct trends with a varying advance coefficient are well-predicted, which indicates the validity of the method used.

In the case of the hybrid mesh, only one operating point, $J = 0.90$, was considered. Using the schemes described earlier, the computations were no longer stable with the hybrid grid, which forced changing the discretizations. The additional instability was traced back to the momentum equation where random errors occurred in the solution process. Thus, a different discretization for the convection of momentum was used: instead of the `limitedLinear 0.33` scheme, the `blended 0.75` was applied. The `blended` scheme is a combination of the linear interpolation and the upwind interpolation. The ratio is fixed so that `blended 0.75` means that 75% of the interpolated value is evaluated with the linear interpolation and 25% with the upwind interpolation [4].

In order to evaluate the performance of the changed settings, the periodic case was recomputed. Force prediction is compared between the two discretizations in Tables 1 and 2. Also the measured data is included. The forces change by approximately 1% from the earlier results. The absolute accuracy of the results is not known so it cannot be stated whether the change is towards better results. Generally, the fixed blending of the two interpolation schemes is expected to produce more numerical error than the one that adjusts to the solution. In addition, the fact that the results change more at low and high advance coefficients supports this view. Near the design point of the propeller (approximately $J = 0.90$), the flow around the propeller is the smoothest and it is probably relatively easy to solve from the computational point of view. Further from the design point the angle of attack changes which also makes the flow more complicated. However, the results are still at an acceptable level in particular for the purposes of this study.

Table 1: Comparison of the thrust coefficient, K_T, computed with the `blended 0.75` scheme ($K_{T,BL}$) to that computed with the `limitedLinear 0.33` ($K_{T,LL}$) and to the measured values ($K_{T,M}$) computed with the periodic grid.

J	Relative difference $\left(\frac{K_{T,BL}-K_{T,LL}}{K_{T,LL}}\right)$ [%]	Relative difference $\left(\frac{K_{T,BL}-K_{T,M}}{K_{T,M}}\right)$ [%]
0.65	0.81	3.50
0.90	0.40	0.64
1.15	0.86	1.24

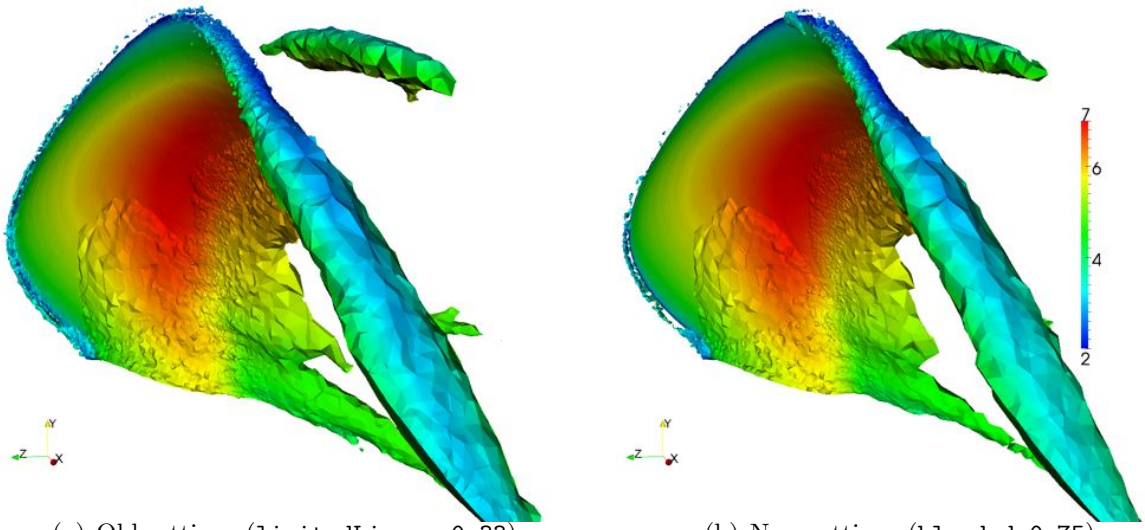
Table 2: Comparison of the torque coefficient, K_Q, computed with the `blended 0.75` scheme ($K_{Q,BL}$) to that computed with the `limitedLinear 0.33` ($K_{Q,LL}$) and to the measured values ($K_{Q,M}$) computed with the periodic grid.

J	Relative difference $\left(\frac{K_{Q,BL}-K_{Q,LL}}{K_{Q,LL}}\right)$ [%]	Relative difference $\left(\frac{K_{Q,BL}-K_{Q,M}}{K_{Q,M}}\right)$ [%]
0.65	1.07	1.22
0.90	0.72	0.72
1.15	1.15	0.39

The effect of the new discretization on the wake field was estimated by inspecting the wing tip vortices emanating from the propeller. Figure 6 shows iso-volumes of vorticity

$$\omega = \nabla \times \mathbf{U} = 300 \ 1/\mathrm{s} \tag{2}$$

that illustrate the tip vortices. With the new settings, the tip vortex is slightly thinner. There is also less vorticity in the central parts of the blade. These observations confirm the expectation that the new settings are more diffusive than the old ones. They are, however, also considerably more robust and still produce propeller performance and wake field well. This encourages to proceed with the new settings.

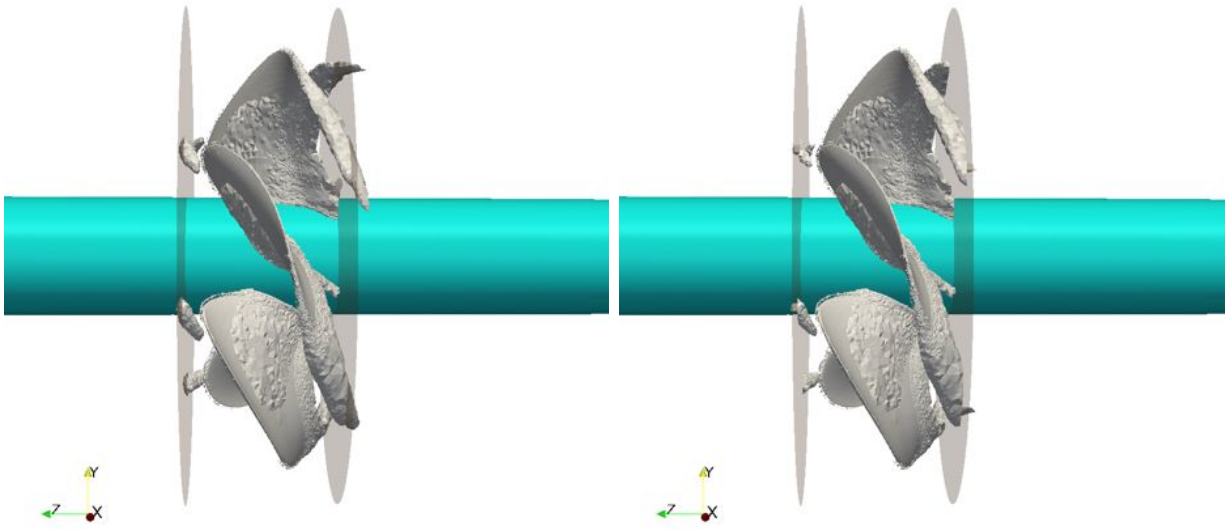


(a) Old settings (`limitedLinear 0.33`) (b) New settings (`blended 0.75`)

Figure 6: Iso-volumes of vorticity $\omega = 300\ \frac{1}{\mathrm{s}}$ in the periodic case with the old and the new settings. Colored by velocity magnitude.

The `MRF` method is further studied with the hybrid grid. Keeping in mind the goal of analyzing multi-propeller devices, the `MRF` cannot always be applied in the entire mesh. The case was computed once with `MRF` applied everywhere and once with `MRF` applied only inside the cylindrical domain around the propeller. As a remark, `MRF` must always be applied inside a region the boundaries of which have no component in the direction of propeller rotation. This condition is fulfilled by any axisymmetric volume such as a cylinder. The requirement comes from the differences in the convective velocities in the cases of `MRF` or no `MRF` .

Figure 7 shows the tip vortices in both cases. Clearly, the vorticity vanishes at the mesh boundary when `MRF` is only applied inside the propeller mesh. In the other case, the field propagates smoothly through the interface. This implies difficulties in applying `MRF` on multiple propellers. Of course, it is also principally wrong to compute a flow around two propellers as a steady-state problem, but should it produce reasonable values, such a method would enable efficient computations, e.g. for R&D purposes. All experiences so far, however, have implied the opposite.



(a) MRF method is applied in domain 2. Outer disturbances are not shown.

(b) MRF method is applied inside the propeller mesh.

Figure 7: Iso-volumes of vorticity with the MRF method applied in two different domains.

4.2 Transient Computations with the Hybrid Grid

Transient computations are conducted in order to determine the influence of parameters related to the temporal discretization on the results. Since the solution procedure iterates a solution within each time step, there are three parameters that are expected to influence the results in the transient case:

- time discretization,
- time step size and
- the number of iterations inside one time step.

The time step defines how rapid phenomena can be reproduced in the computation. Clearly, events shorter or even the length of the time step cannot be predicted in the computation. The time discretization directly affects the truncation error present in the computation and plays thus a clear role in the accuracy of a time accurate computation. Finally, the number of iterations conducted inside a time step also affects the solution. The iteration process inside a time step depends on the two other parameters and the problem of finding suitable settings can only be solved by studying how the three parameters interact.

The parameters were varied and the resulting force coefficients are plotted against time in Figure (8). In each subfigure, the measured value is given as a solid horizontal line and the steady-state result is represented by the dashed horizontal line. The steady-state results are those computed with MRF applied in the entire grid.

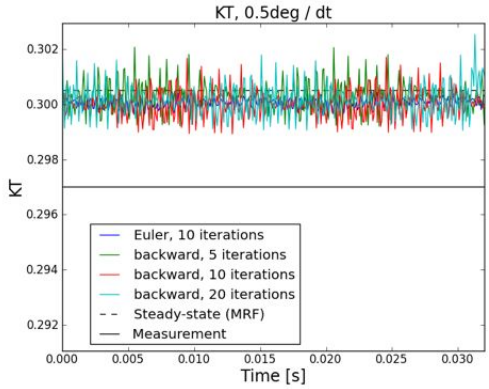
The first two subfigures illustrate the results with the smallest time step corresponding to 0.5° of propeller rotation per time step. Maximum Courant number in these cases is approximately $Co \approx 95$; the maximum value being located at the propeller tip. Both time discretizations yield the same average results. The backward-scheme gives more oscillations which is probably due to its lower diffusivity as compared to the Euler method. The number of iterations inside a time step does not affect the average results either, which means that the error due to time discretization is not dominant compared to other factors. A further implication of a sufficiently accurate temporal discretization is the fact that the results are very close to the steady-state results computed with MRF applied in the entire grid. These results are referred to as *temporally converged* results in the following.

In the case of 2° rotation per time step (maximum $Co \approx 375$), shown in the next two subfigures, the time discretization plays a clear role. With an increasing number of iterations inside a time step, the results approach the level obtained with the smaller time step. Using ten iterations yields results that differ from the *temporally converged* results by approximately 2%. With 20 iterations, the values are closer and with 40 iterations the results are the same as with the smaller time step. Thus it seems that the numerical error due to time discretization does not play a significant role even with 2 degrees of rotation per time step, but the error due to insufficient iteration (*iterative* error) does affect the solution.

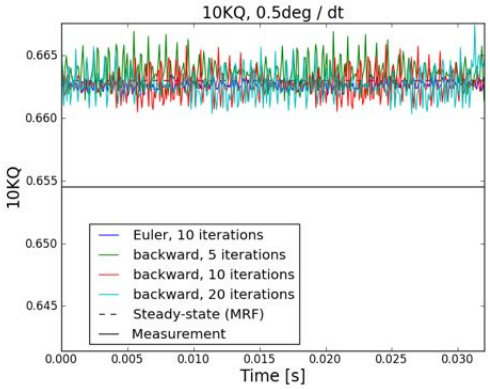
If other parameters are kept constant the backward discretization predicts the forces better than the Euler discretization. This could be expected, since the truncation error of the backward scheme is theoretically lower than that of the Euler scheme. According to the conclusion that only the iterative error affects the forces with the propeller rotation of $(2°/\Delta t)$, not the numerical error, the differences in the numerical errors should not explain the difference, however. Based on these results, the backward scheme seems to reduce the iterative error, too, compared to the Euler method. Thus, the results encourage using the backward scheme, if only there is enough memory available.

The lowest subfigures illustrate the results with the three time step sizes applied. As mentioned earlier, the two smallest time steps yield the same average values if enough iterations are used inside a time step. A time step corresponding to a rotation of 5° clearly under-predicts the forces also with 40 inner iterations. Other tests not reported here implied that even a larger number of iterations would probably not yield the temporally converged results, but it has not been thoroughly tested.

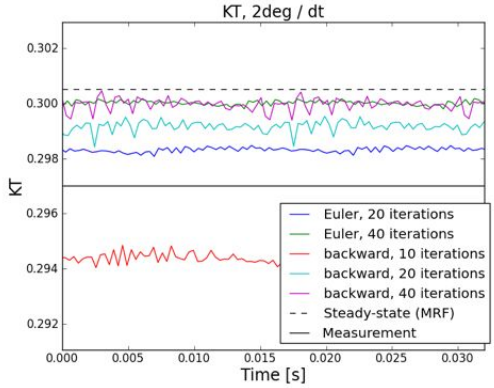
A correct analysis of multiple propellers requires not only the forces to be predicted correctly but also the wake field. Contours of tangential velocities are shown in Figure 9. It seems that the larger the time step the sharper the peaks are in the wake field. Furthermore, the fewer iterations are conducted within a time step, the sharper the peaks



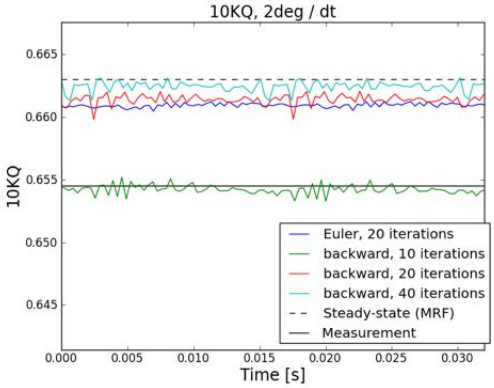
(a) Thrust coefficient K_T. Mesh movement per time step corresponds to 0.5°.



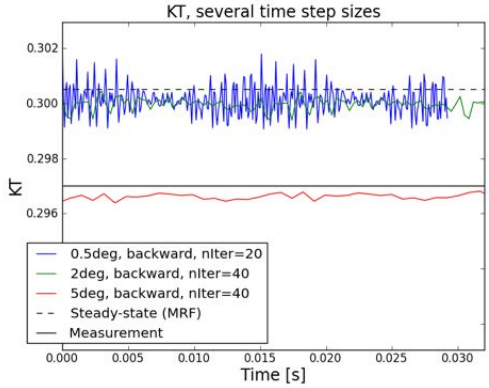
(b) Torque coefficient K_Q. Mesh movement per time step corresponds to 0.5°.



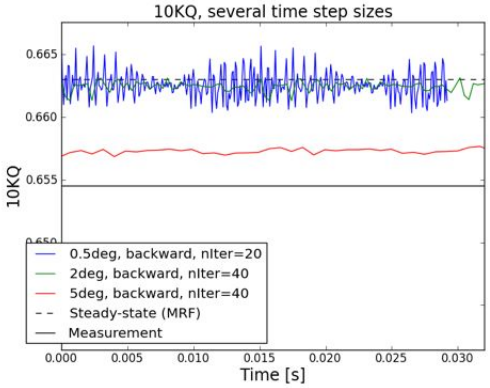
(c) Thrust coefficient K_T. Mesh movement per time step corresponds to 2°.



(d) Torque coefficient K_Q. Mesh movement per time step corresponds to 2°.



(e) Thrust coefficient K_T. Effect of the time step size.



(f) Torque coefficient K_Q. Effect of the time step size.

Figure 8: Force coefficients (K_T and K_Q) against time at $J = 0.90$.

are in the wake. It seems that a more accurate solution in time smoothes out the velocity peaks in the solution. This was not expected, since accuracy often relates to a less diffusive scheme and less mixing in the solution. The present results clearly show that in the prediction of the wake, the opposite happens. The only case where less accuracy yields a smoother wake field is if one compares the `backward` and `Euler` methods with each other.

Perhaps the most important finding here is the fact that the two cases with a rotation of 0.5° per time step do not yield the same wake fields. It is important, since the force prediction was the same in both cases. They were both able to reproduce the temporally converged forces. Thus, the force prediction is not sufficient to guarantee a correct overall flow.

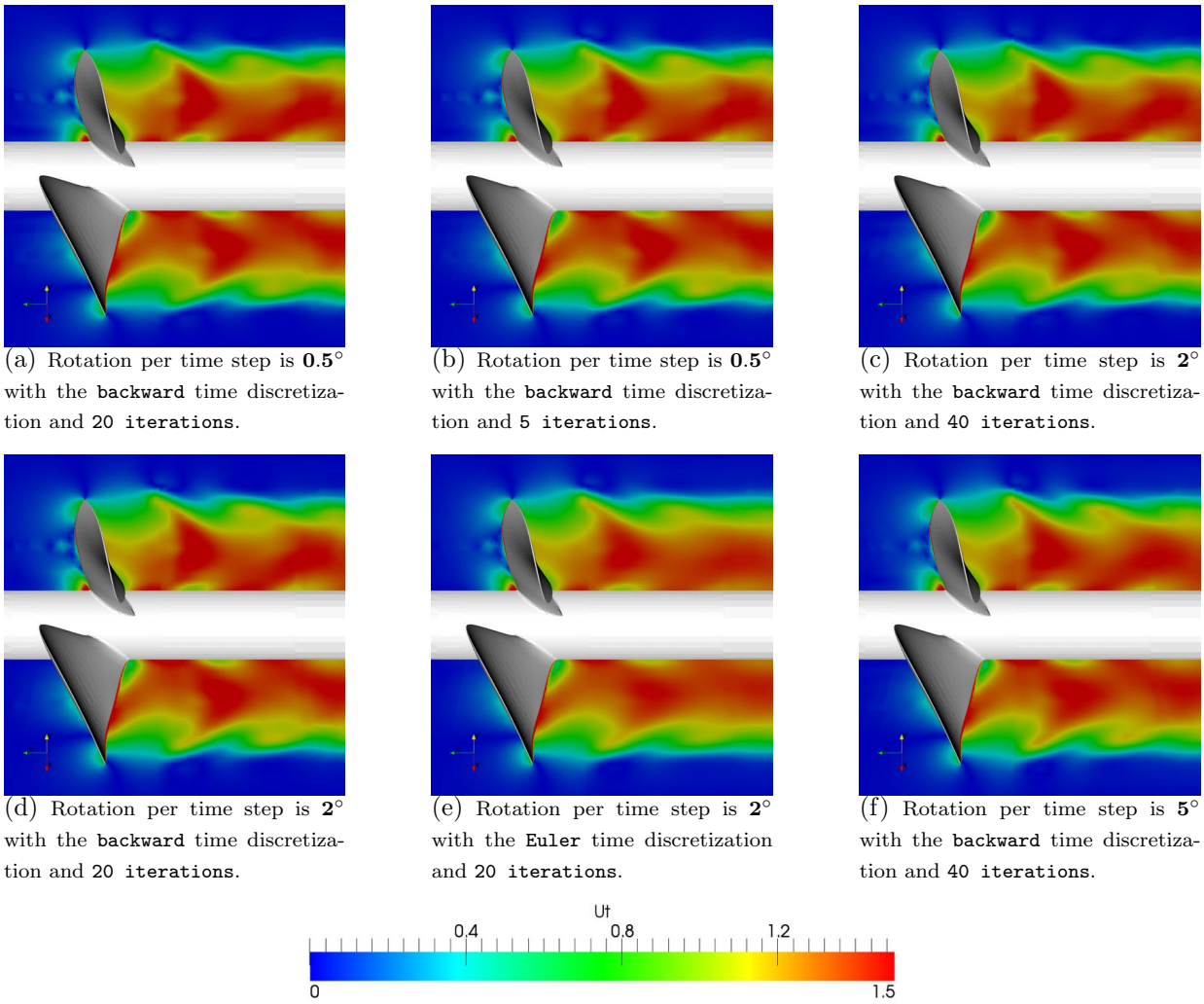
5 Conclusions

A marine propeller was analyzed as a steady-state and a time dependent problem in order to evaluate the performance of the computational methods available in `OpenFOAM`. Two types of meshes were utilized. One was a continuous mesh and the other consisted of two parts that were connected by a non-conformal mesh interface. Turbulence was modelled with the SST k-ω model and the y^+ values on the blade surface were approximately $y^+ \approx 1$. Both global propeller forces and the propeller wake field were analyzed. The most important findings are listed in the following.

- The `simpleFoam` code predicts forces well in the case of a single propeller with `MRF`
- A mesh discontinuity impairs the stability of the computations. This is fixed by using the `blended 0.75` interpolation for the convection of momentum
- The scheme seems to yield results in computations with engineering accuracy
- Applying `MRF` with more than one propellers will involve difficulties
- Transient code produces results very close to those predicted by the steady-state computations
- Too few iterations within a time step can impair the solution considerably
- The requried number of iterations per time step strongly depends on time step size
- Prediction of the wake field requires more accurate methods than force prediction

Acknowledgements

Simulations have been done using taito.csc.fi HP super cluster at CSC - IT Center for Science Ltd.



(a) Rotation per time step is **0.5**° with the `backward` time discretization and `20 iterations`.

(b) Rotation per time step is **0.5**° with the `backward` time discretization and `5 iterations`.

(c) Rotation per time step is **2**° with the `backward` time discretization and `40 iterations`.

(d) Rotation per time step is **2**° with the `backward` time discretization and `20 iterations`.

(e) Rotation per time step is **2**° with the `Euler` time discretization and `20 iterations`.

(f) Rotation per time step is **5**° with the `backward` time discretization and `40 iterations`.

Figure 9: Propagation of tangential velocities behind the propeller.

REFERENCES

[1] Menter, F., Esch, T., *Elements of Industrial Heat Transfer Prediction*, 16th Brazilian Congress of Mechanical Engineering (COBEM), Nov. 2001.

[2] `www.openfoam.org`

[3] Carlton J. Marine Propellers and Propulsion, Second edition, Elsevier Ltd, 2007, ISBN: 978-07506-8150-6.

[4] `OpenFOAM Programmer's Guide, Version 2.2.0, 22nd February, 2013.`

[5] Farrell, P. E., Maddison, J. R., Conservative interpolation between volume meshes by local Galerkin projection, Comput. Methods Appl. Mech Engrg 200:89 (2011).